MODE(1) There are other ways of handling designs in MODE(1). One way, often used for complicated designe, is to put the X and Y co-ordinates into lines of DATA and using READ. Here is an example of such a program.

```
0 'MICRO MAGIC
1 'MAP OF AUSTRALIA
10 MODE(1):COLOR7,0:Y=12
11 B=2:GOSUB100:B=4:GOSUB100:B=11:GOSUB100:B=14:GOSUB100
12 B=11:GOSUB100:B=11:GOSUB100:B=11:GOSUB100:B=7:GOSUB100
13 B=12:GOSUB100:B=12:GOSUB100:B=9:GOSUB100:B=10:GOSUB100
14 B=6:GOSUB100:B=14:GOSUB100:B=5:GOSUB100:B=4:GOSUB100
15 B=5:GOSUB100:B=4:GOSUB100:B=3:GOSUB100:B=2:GOSUB100
16 B=3:GOSUB100:B=2:GOSUB100:B=4:GOSUB100:B=3:GOSUB100
17 B=6:GOSUB100:B=5:GOSUB100:B=5:GOSUB100:B=5:GOSUB100
18 B=5:GOSUB100:B=4:GOSUB100:B=20:GOSUB100:B=14:GOSUB100
19 B=17:GOSUB100:B=14:GOSUB100:B=12:GOSUB100:B=16:GOSUB100
20 B=9:GOSUB100:B=2:GOSUB100:B=2:GOSUB100:B=9:GOSUB100
21 B=5:GOSUB100:B=2:GOSUB100:B=2:GOSUB100:B=3:GOSUB100
22 B=5:GOSUB100:B=3:GOSUB100
23 GOTO50

30 DATA86,87,57,58,86,88,55,56,57,60,61,62,69,71,85,86,88,60,61
31 DATA62,63,64,65,66,67,68,69,70,85,86,89,56,57,58,59,60,68,69
32 DATA70,85,89,90,49,50,51,52,53,56,57,68,85,89,90,91,47,48,49
33 DATA53,54,55,67,70,85,90,91,46,47,67,84,85,90,91,45,46,67,68
34 DATA69,70,71,83,84,91,92,93,41,42,43,44,45,71,72,73,74,75,83
35 DATA93,39,40,41,75,76,77,82,83,93,37,38,39,77,78,79,80,81,82
36 DATA93,33,34,35,36,37,93,22,24,25,26,27,28,29,30,31,32,33,93
37 DATA94,95,21,22,23,24,95,21,22,95,96,20,21,96,97,98,19,20,98
38 DATA99,19,99,19,99,19,99,100,19,100,17,19,20,100,18,20,100
39 DATA19,20,21,98,99,100,20,21,97,98,99,21,22,96,97,98,22,23
40 DATA94,95,96,23,24,25,93,94,24,25,26,93,26,27,47,48,49,50,51
41 DATA52,53,54,55,56,57,58,59,60,61,62,63,93,27,43,44,45,46,47
42 DATA63,64,65,66,73,74,75,93,26,27,37,38,39,40,41,42,43,66,67
43 DATA68,72,73,75,93,94,27,28,35,36,37,68,69,70,71,72,75,91,92
44 DATA93,28,29,33,34,35,70,71,74,75,89,90,91,29,30,31,32,33,75
45 DATA76,77,78,79,80,84,85,87,88,89,69,70,76,80,81,82,83,84,86
46 DATA84,88,76,78,78,79,80,81,85,86,87,88,89,78,82,83,84,88,79
47 DATA88,80,88,80,81,87,81,84,85,86,87,82,83,84

50 COLOR6,0
51 Y=19:FORI=1TO23:SET(54,Y):Y=Y+1:NEXT
52 X=55:FORI=1TO26:SET(X,31):X=X+1:NEXT
53 Y=23:FORI=1TO8:SET(76,Y):Y=Y+1:NEXT
54 Y=31:FORI=1TO16:SET(80,Y):Y=Y+1:NEXT
55 X=81:FORI=1TO12:SET(X,34):X=X+1:NEXT
56 X=92:FORI=1TO5:SET(X,33):X=X+1:NEXT
57 X=96:FORI=1TO3:SET(X,34):X=X+1:NEXT
58 SET(98,33):SET(99,33)
59 SET(80,41):SET(81,41):SET(82,41):SET(89,43):SET(90,43)
60 SET(91,43):SET(92,43)
61 X=83:FORI=1TO7:SET(X,42):X=X+1:NEXT
62 GOTO62

100 FORA=1TOB:READX:SET(X,Y):NEXT:Y=Y+1:RETURN
```

You will appreciate that masses of DATA need to be compliled carefully, testing line-by-line, as to DEBUG a completed program could be quite confusing.

And then there's POKE...

A study of the MODE(1) VIDEO DISPLAY WORKSHEET (Screen Grid) will reveal an important fact. Looking at the corners, you'll see the Memory Locations, which are given in HEXIDECIMAL and DECIMAL (in brackets), refer to 4 pixels, not one.
The Memory Addresses for POKEs are the Decimal locations.
If we were to POKE 28672 with any number between Ø and 255 then the four top lefthand pixels would be changed. Some numbers POKEd into 28672 would "turn on" some of the pixels in a color leaving the rest "turned off". Other numbers would turn on pixels in different colors. There are 256 permutations.
Here is a program for drawing borders as in MICROMAZE...

```
Ø PROGRAM FOR MODE(1) BORDERS.
1ØØ MODE(1):COLOR3,1
11Ø S=28672:GOSUB2ØØ:S=S+32:GOSUB2ØØ
12Ø S=30624:GOSUB2ØØ:S=S+32:GOSUB2ØØ
13Ø S=28736:GOSUB21Ø:S=S+31:GOSUB21Ø
14Ø GOTO14Ø
2ØØ FORI=STOS+31:POKEI,17Ø:NEXT:RETURN
21Ø FORI=STOS+192ØSTEP32:POKEI,17Ø:NEXT:RETURN
```

Here is a program that you may like to experiment with. It prints a number on the screen and then changes to MODE(1) and shows you the effect of POKING that number into a MODE(1) display. By pressing P the next number is printed and then displayed.

```
100 CLS:CLEAR:FORP=0TO255:PRINT@270,P:FORT=1TO500:NEXT
110 MODE(1):S=29612
120 FORI=1TO8:FORJ=STOS+8:POKEJ,P:NEXT:S=S+32:NEXT
130 Z$=INKEY$:Z$=""
140 Z$=INKEY$:IFZ$=""THEN140
150 IFZ$="P"THENNEXT
160 GOTO140
```

This next program allows you to INPUT Ø to 255 which will be POKEd into MODE(1) screen locations.

```
100 CLS:CLEAR:COLOR,1
110 PRINT@235,"";:INPUTP:MODE(1):S=29612
120 FORI=1TO8:FORJ=STOS+8:POKEJ,P:NEXT:S=S+32:NEXT
130 Z$=INKEY$:Z$=""
140 Z$=INKEY$:IFZ$=""THEN140
150 IFZ$="P"THEN110
160 GOTO140
```

Here is a TABLE derived from the above programs...

POKE	COLOR,Ø	COLOR,1	
,Ø	GREEN	BUFF	(BACKGROUND COLORS)
,85	YELLOW	CYAN	
,17Ø	BLUE	MAGENTA	
,255	RED	ORANGE	

...and the advantage of POKEs over SETs is SPEED!

GREMLINS The Veeled Gremlins allow obscure errors to be found immediately, but the glaringly obvious ones defy detection by becoming invisible. In MICRO MAGIC No.2 there were a couple of the invisible sort.

Page 6. Line 52Ø should read 52Ø PRINT@32,"DELETE":INPUTD$
Page 6. Line 54Ø should read 54Ø IFD$=""THEN23Ø (the line number was missing).
Page 8. Line 43Ø should read 42Ø IFA=126THENA-Ø:GOSUB8ØØ
Page 4. Line 28Ø can be deleted completely. It duplicates Line 23Ø.
Also, if you add a lot of IF statements you could add GOTO44Ø at the end of each line. In MICRO TYPE you could put GOTO44Ø at the end of lines 39Ø, 4ØØ and 41Ø, but hardly worth it with so few IFs.

SORRY 'BOUT THAT...

```
10 '*************************
20 '* SPRITE * AD LIB VEEZED *
30 '**********  MICRO CLUB   *
40 '          *****************
    Underlines characters are in INVERSE.
1000 CLEAR800:COLOR,1:CC=1:GOSUB9000:GOTO3000
2000 GOSUB9000
2010 PRINT"  1. DRAW CHARACTER":PRINT"  2. EDIT CHARACTER"
2020 PRINT"  3. DISPLAY CHARACTER":PRINT"  4. SAVE CHARACTER(S)"
2030 PRINT"  5. LOAD CHARACTER(S)":PRINT"  6. END":PRINT
2040 PRINT"ENTER YOUR CHOICE.";
2050 INPUTC:IFC<1ORC>6THENC=0
2060 IFC=1THEN4000ELSEIFC=2THEN5000ELSEIFC=3THEN6000
2070 IFC=4THEN7000ELSEIFC=5THEN8000ELSEIFC=6THENEND
2510 CLS:PRINT" RE-ENTER.":GOTO2040
3000 PRINT"ENTER THE NUMBER OF CHARACTERS   YOU WANT TO SET. ";
3010 PRINT"THE MAXIMUM IS 10.";:INPUTN
3030 IFN<1ORN>10THENPRINT" PLEASE RE-ENTER.":GOTO3000
3040 PRINT:PRINT" ENTER MATRIX SIZE OF CHARACTER."
3050 PRINT:PRINT" MAXIMUM IS 8 X 8"
3060 PRINT:INPUT" ACROSS ";X:INPUT" DOWN   ";Y
3070 IFX>8ORY>8THENCLS:PRINT" PLEASE RE-ENTER.":GOTO3040
3100 GOSUB9600:N=N-1:X=X-1:Y=Y-1:DIMT%(N,X,Y)
3110 FORI=0TON:FORJ=0TOX:FORK=0TOY:T%(I,J,K)=1
3120 NEXT:NEXT:NEXT:GOTO2000
4000 IFCN=N+1THEN PRINT"NO NEW CHARACTERS LEFT!":GOTO4100
4010 Z=CN:D=0:A=0:CLS:PRINT@18,"DRAW OPTION":COLOR1
4020 FORI=0TO(Y)*32STEP32:FORJ=0TOX:PRINT@1+J,"■";:NEXT:NEXT
4030 GOSUB9100:GOSUB9200:CN=CN+1:GOTO2000
4100 INPUT"PRESS RETURN";C$:GOTO2000
5000 GOSUB9400
5030 D=0:A=0::CLS:PRINT@18, "EDIT OPTION"
5040 FORI=0TOY:FORJ=0TOX:COLORT%(Z,J,I):PRINT@I*32+J,"■";
5050 NEXT:NEXT:GOSUB9100:GOSUB9200:GOTO2000
6000 CLS:PRINT" PRESS THE SPACE KEY TO EXIT":PRINT:GOSUB9400
6100 MODE(1):FORI=0TOY:FORJ=0TOX:COLORT%(Z,J,I):SET(J,I):NEXT
6105 NEXT
6120 C$=INKEY$:IFC$=" "THEN2000ELSE6120
7000 CLS:PRINT@9,"CHARACTER SAVE"
7010 PRINT:GOSUB9600:PRINT" WHILE I SET UP."
7020 FORI=0TON:A$(I)=MID$(STR$(X),2,1)+MID$(STR$(Y),2,1):NEXT
7030 FORI=0TON:FORJ=0TOX:FORK=0TOY
7040 A$(I)=A$(I)+MID$(STR$(T%(I,J,K)),2,1):NEXT:NEXT:NEXT
7070 GOSUB9500:GOSUB9600:FORI=0TON:PRINT#"CHAR",A$(I):NEXT
7080 GOTO2000
8000 CLS:PRINT@9,"CHARACTER LOAD":PRINT:GOSUB9500:GOSUB9600
8010 FORI=0TON:INPUT#"CHAR",A$(I):NEXT
8020 X=VAL(LEFT$(A$(0),1)):Y=VAL(MID$(A$(0),2,1))
8030 FORI=0TON:A$(I)=RIGHT$(A$(I),LEN(A$(I))-2)
8040 FORJ=0TOX:FORK=0TOY:T%(I,J,K)=VAL(LEFT$(A$(I),1))
8045 IFLEN(A$(I))=1THEN8060
8050 B$=RIGHT$(A$(I),LEN(A$(I))-1):A$(I)=B$
8060 NEXT:NEXT:NEXT:GOTO2000
9000 CLS:PRINT@12,"SPRITE":PRINT:RETURN
9100 PRINT@81,"CONTROL KEYS";:PRINT@145,"LEFT  = M";
9110 PRINT@177,"RIGHT = ,";:PRINT@209,"UP   = .";
9120 PRINT@241,"DOWN  = <SPACE>";:PRINT@273,"SET   = L";
9130 PRINT@305,"COLOR = 1 TO 4";:PRINT@337,"EXIT  = E";
9140 PRINT@401,"COMMAND?";:RETURN
9200 PRINT@D*32+A," ";
9204 FORI=1TO25:C$=INKEY$:NEXT:PRINT@412,C$;
9206 COLORT%(Z,A,D):PRINT@D*32+A,"■";:COLORCC
9210 IFC$="M"ANDA>0THENA=A-1
9220 IFC$=","ANDA<XTHENA=A+1
9230 IFC$="."ANDD>0THEND=D-1
9240 IFC$=" "ANDD<YTHEND=D+1
9250 IFC$="1"THENCOLOR1:CC=1
9260 IFC$="2"THENCOLOR2:CC=2
9270 IFC$="3"THENCOLOR3:CC=3
9280 IFC$="4"THENCOLOR4:CC=4
9290 IFC$="L"THENPRINT@D*32+A,"■";:T%(Z,A,D)=CC
9300 IFC$="E"THENRETURN
9310 GOTO9200
9400 INPUT" ENTER CHARACTER NUMBER";Z
9410 IFZ<1ORZ>N+1THENPRINT" PLEASE RE-ENTER":GOTO9400
9420 Z=Z-1:RETURN
9500 PRINT" PLEASE SET UP TAPE.":PRINT" THEN PRESS <RETURN>."
9510 INPUTC$:RETURN
9600 PRINT"      STAND-BY":RETURN
```

SPRITE allows you to design up to 10 characters. After selecting the number of characters you want to design, you are asked to select the MATRIX. A matrix is a grid and can be up to 8 squares across, and up to 8 squares down. Your design could be, for instance on a matrix 7 X 5, like this.....

```
[┬┬┬┬┬┬┬]
[┬┬┬┬┬┬┬]
[┬┬┬┬┬┬┬]
[┬┬┬┬┬┬┬]
[┴┴┴┴┴┴┴]
 7 X 5 MATRIX
```

The selected matrix is displayed in the background color in MODE(0) with a flashing cursor, which can be moved around the squares of the grid. You can select any one of 4 colors for each square to make up your design. It might be a little man, a space ship or an aircraft. When a design is completed press E to lock the design into memory and return to the menu. If you select DISPLAY CHARACTER you are asked to enter the design number (from 1 to 10) and that design will be shown in MODE(1). Easy? Of course!

The sub-routines for SAVING and LOADING the designs can be written into games programs. Care must be taken to preserve the variables in other programs. This program needs a color TV for best effect.